

Appln. No. 09/884,577
Response Dated July 19, 2004
Office Action dated March 18, 2004
Docket No. 6169-227

IBM Docket No.: BOC9-2000-0094

REMARKS/ARGUMENTS

These remarks are made in response to the Office Action of March 18, 2004 (Office Action). This response is being filed with a petition for a one month retro-active extension of time with the appropriate fee.

In paragraphs 2-3, the Examiner has rejected claims 1-8, and 10-15 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,243,453 to Bunch, *et al.* (Bunch). In paragraphs 4-5, the Examiner has rejected claim 9 under 35 U.S.C. § 103(a) as being unpatentable over Bunch in view of U.S. Patent 5,991,803 to Glitho, *et al.* (Glitho).

Prior to addressing the rejections on the art, a brief review of the Applicants' invention is in order. The Applicants' claimed and disclosed subject matter teaches a service logic execution environment (SLEE) for use in an intelligent network model having an application layer and a protocol layer. For example, the SLEE can implement a JAIN Service Logic Execution Environment (JSLEE) interface. The present invention solves the deficiencies of the prior art by providing more direct processing of events by the service components and by reducing protocol stack specific code contained in the SLEE. A SLEE which has been configured in accordance with the inventive arrangements can include a class loader for loading service components in the SLEE, and an event routing bus for receiving events from the protocol layer and other service components.

Turning specifically to the rejections on the art, in paragraphs 2-3, the Examiner has rejected claims 1-8, and 10-15 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,243,453 to Bunch, *et al.* (Bunch). Bunch teaches how to construct a programmable call processing system for an AIN.

Referring to claim 1, Applicants claim:

1. A service logic execution environment (SLEE) in an intelligent network model, said network model comprising an application layer and a protocol layer, said SLEE comprising:

a class loader for loading service components in the SLEE, the SLEE registering each loaded service component to receive events directed to particular registered service components; and,

an event routing bus for receiving events from the protocol layer and other service components, said event routing bus routing said received

Appln. No. 09/884,577
Response Dated July 19, 2004
Office Action dated March 18, 2004
Docket No. 6169-227

IBM Docket No.: BOC9-2000-0094

events to said particular registered service components executing in the SLEE.

Applicants emphasize claim 1 is explicitly directed towards a SLEE. Specifically, the claimed invention details a way to fabricate a SLEE so that service components can receive and process events from other service components and external applications, as stated on lines 6-8 of page 9. Accordingly, the Applicants overcome limitations of conventional implementations of a SLEE (shown in Figure 1) where service components can only receive and process events received from a protocol stack via the SLEE.

Bunch does not explain how to improve the traditional structure of a SLEE in any fashion. Specifically, the (SCP 18) is not equivalent to the SLEE of FIG. 2. Additionally, the SCE 14 is not equivalent to the class loader 212. In fact, Bunch explicitly utilizes "a standard call processing process performing call processing according to industry standard call models," as stated at column 1, lines 46-47. Figure 4 of Bunch (item 82) show that Bunch relies upon a standard system call processing structure to provide call logic services. In contrast, Applicants teach a system that does not utilize standard call processing, as shown in FIG. 2.

Moreover, FIG. 3 of Bunch explicitly teaches that APIs (items 56, 62, 64) are exposed directly to the services being provided (services 58, 60, 66). As detailed in the background of the Applicants' invention (and in FIG 1), directly exposing services to APIs is "the old way" of providing services. In contrast, the Applicants teach a system that divorces call control (and other APIs) from the services being built, as shown in FIG. 2 and explicitly claimed by the "SLEE registering each loaded service component to receive events directed to particular registered service components."

Additionally, Applicants specifically claim an event bus 204 that threads services and applications together. This concept is missing from Bunch. That is, the SSP or MSP 10 fails to teach "an event routing bus for receiving events from a protocol layer and other service components that routes the received events to particular registered service components." Bunch does not teach registering service components, routing events to service components (except through an API which Applicants teach away from) or anything resembling the Applicants' claimed invention.

Appln. No. 09/884,577
Response Dated July 19, 2004
Office Action dated March 18, 2004
Docket No. 6169-227

IBM Docket No.: BOC9-2000-0094

In light of the above, Bunch clearly fails to teach limitations explicitly claimed by the Applicants in claim 1. Accordingly, the 35 U.S.C. § 102(e) rejection as to claims 1-8 should be withdrawn, which action is respectfully requested.

Referring to claims 10 and 13, Applicants claim the steps of:

receiving at least one event from a service component executing in the SLEE; and,
routing each received event to a service component which has registered with the SLEE to receive said routed event.

As previously noted, Bunch fails to teach registering service components with a SLEE, which is claimed by the Applicants above. Bunch teaches that APIs (items 56, 62, 64) are exposed directly to the services being provided (services 58, 60, 66). Accordingly, Bunch's teachings directly contradict a method where "at least one event from a service component" can be "routed to a service component that has registered to receive the routed event." Consequently, Bunch fails to teach claim 10 and/or claim 13. Therefore, the 35 U.S.C. § 102(e) rejection as to claims 10-15 should be withdrawn, which action is respectfully requested.

In paragraphs 4-5, the Examiner has rejected claim 9 under 35 U.S.C. § 103(a) as being unpatentable over Bunch in view of U.S. Patent 5,991,803 to Glitho, *et al.* (Glitho). Glitho fails to cure the deficiencies of Bunch.

Glitho teaches how to decouple service creation from service execution by building services out of generic components. These generic components are transformed by a service execution agent into executable components for a SLEE. The service execution agent of Glitho serves as a compiler that translates generic code in to SLEE specific code. This means that the service execution agent must be changed whenever the SLEE changes.

In contrast, the Applicants do not teach that services should be built out of generic code. The Applicants solution does not depend upon a service execution agent or its equivalent. Instead, the Applicants utilities JAVA to develop interfaces to isolate services from implementation specifics of underlying protocols. Consequently, the approach taught by Glitho is counter to the approach taught by the Applicants. That is, Applicants put the life cycle care of services into the SLEE and isolates services running in the SLEE from the protocols. The opposite is taught by Glitho.

Appln. No. 09/884,577
Response Dated July 19, 2004
Office Action dated March 18, 2004
Docket No. 6169-227

IBM Docket No.: BOC9-2000-0094

To summarize, neither Bunch, Glitho, nor any combination thereof teach or suggest a methodology applicable to SLEEs that divorces call control (and other APIs) from the services being built. Neither Bunch, Glitho, nor any combination thereof teach or suggest an event bus 204 that threads services and applications together. Neither Bunch, Glitho, nor any combination thereof teach or suggest registering service components with a SLEE. Further, neither Bunch, Glitho, nor any combination thereof teach or suggest that registered services can request notification of specific events (defined during registration). Consequently, the 35 U.S.C. § 103(a) rejection of claim 9 citing Brunch in view of Glitho is improper. Therefore, Applicants respectfully request that the 35 U.S.C. §103(a) to claim 9 be withdrawn.

Applicants believe that this application is now in full condition for allowance, which action is respectfully requested. Applicants request that the Examiner call the undersigned if clarification is needed on any matter within this Amendment, or if the Examiner believes a telephone interview would expedite the prosecution of the subject application to completion.

Respectfully submitted,

Date: 19 July 2004

Gregory A. Nelson, Registration No. 30,577
Brian K. Buchheit, Registration No. 52,667
AKERMANN SENTERFITT
Post Office Box 3188
West Palm Beach, FL 33402-3188
Telephone: (561) 653-5000